

Operációs rendszerek alapjai

10. gyakorlat

Mai gyakorlat

- Extended regular expression
- Gyakorlás

Extended regular expression (ERE)

- Az eddig ismert reguláris kifejezésekkel csak keresni tudtunk
 - Szeretnénk
 - a reguláris kifejezésben használni a találatot
 - szerkesztéseket is végrehajtani a megtalált objektumokon
 - például Python főverziószámot váltunk
- ```
print "Hello World!" -> print("Hello World")
```
- A megoldás az ERE használata
  - Legtöbb program képes rá a -E kapcsolóval (sed, grep stb.)
  - Legtöbb programozási nyelv tud (E)RE-ket kezelni
  - Legtöbb (jó) szövegszerkesztő tud (E)RE-ket kezelni

# Backreference

- A `()` operátor segítségével eddig mintázatok unióját vettük (grouping)
  - `(a|b)+` megtalálja a 'a' és 'b' karakterek kombinációjával létrehozott nem üres szavakat
- Az operátor ennél többet csinál, el is menti a találatot (capturing)
- Az n-edik csoport találatára `\n`-ként hivatkozhatunk (backreference)
  - backreference-re is vonatkoznak a módosítók
  - példa
    - találat regex: `^print (".*")$`
    - helyettesítő regex: `print (\1)`
- [Ezen](#) a honlapon lehet nagyon jól gyakorolni

# G10F01

- Próbáld ki a Python főverziószámos példát!
  - A honlapon van lehetőség helyettesítésre is

# G10F02

- Tervezz egy reguláris kifejezést, ami megtalálja az összes HTML címsort!

# G10F03

- A MATLAB nyelv már képes kezelni összetett értékadás operátorokat
  - igazából nem
- Tehát `bar=bar+foo` helyett `bar+=foo`
- Tervezz egy reguláris kifejezést, ami elvégzi a helyettesítést!

# G10F04

- Írj egy script-t, ami run-length kódolja/dekódolja a bemenetet!
  - ismétlődő karakterek helyettesítése az ismétlés számával, illetve egy karakter példánnyal
    - AABBB <-> 2A3B
- Kódolás esetén a bemenet csak betűkből állhat!
  - `^[a-zA-Z]+$`
- Dekódolás esetén a bemenet számmal kezdődik, betűvel végződik és minden két szám között van betű.
  - `^([0-9]+[a-zA-Z])+$`
- Implementáció során alkalmazzuk a bash előnyeit!
  - ne karakterenként dolgozzunk, inkább használjunk extended regex-t
  - ha mégis karakterenként, akkor érdemes awk-ban bash helyett
  - irányadó fájl méret: egy-egy sor a kódolás és dekódolás



# G10F05

- Becsüld meg  $\pi$  értékét Monte Carlo módszerrel!
- Lépések, például egy awk script-ben
  - generálj sok (mondjuk  $10^7$ ) random  $(x,y)$  párt
  - számold meg azokat az  $(x,y)$  párokat, amikre  $x^2+y^2 \leq 1$
  - $\pi$  jó közelítéssel  $4 \times \text{count} / 10^7$
- Most párhuzamosítsd!
  - indítsd el mondjuk 100 szálat, amik csak  $10^5$  párt generálnak
    - & jellel háttérben indul el a folyamat, `wait` kulcsszóval megvárja a folyamatokat
  - átlagold az eredményüket
- Hasonlítsd össze a futásidőket!

# Szorgalmi HF03 (15 pont)

- Írj párhuzamosított merge sort-t egész számokra!
  - bemeneti struktúra tetszőleges (csv, tsv stb.)
- Tetszőleges nyelv használható a gyakorlaton tanultak közül
- Javasolt struktúra
  - rendező awk script
  - összefűző awk script
    - feltételezheted, hogy két rendezett tömb a bemenete
  - bash script
    - meghívja a rendező awk script-t a tömb két felére
    - kimenetekre meghívja az összefűző awk script-t
- Hasonlítsd össze a futásidőt a soros verzióval!

# Érdekességek, olvasnivalók

- [Regex gyakorló honlap](#)
- [Uncle Bob: Clean Code lessons](#)
- [Uncle Bob: The Future of Programming](#)